U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute, Affiliated to VTU

June 2025 Semester End Main Examinations

Programme: B.E.	**Semester: IV**
Branch: Computer Science and Business Systems	**Duration: 3 hrs.**
Course Code: 23BS4PCADA	**Max Marks: 100**
Course: Analysis and Design of Algorithms	

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

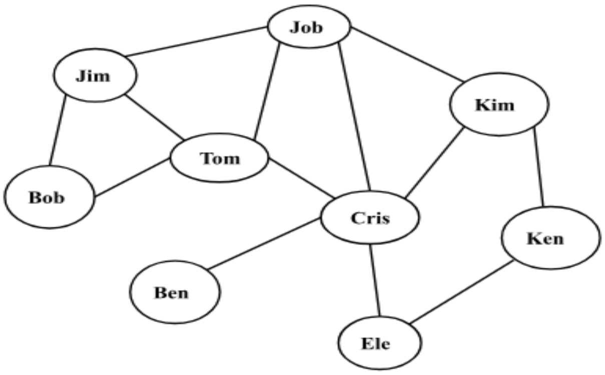
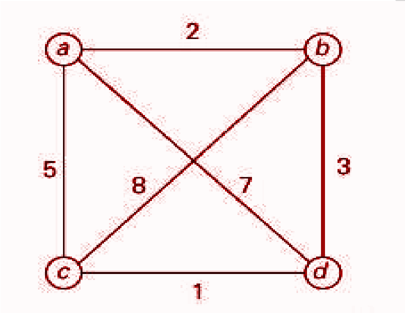
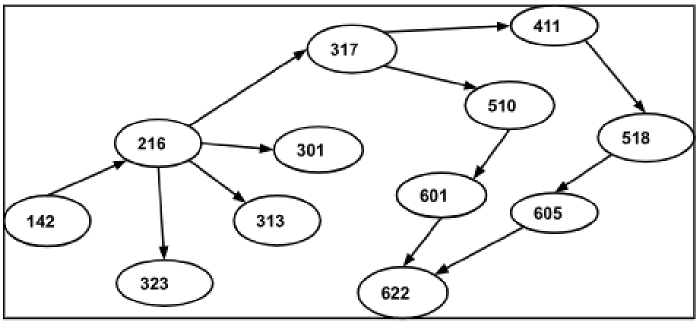
Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

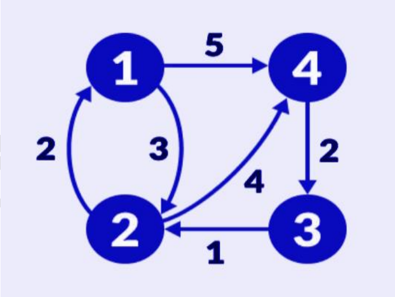
		UNIT - I	***CO***	***PO***	**Marks**
1	a)	Elucidate the algorithm design and analysis process with a neat diagram.	*CO1*	*PO1*	**5**
	b)	Define and explain asymptotic notations with appropriate graphical representation. Compare and contrast Big-O, Omega, and Theta notations with examples.	*CO2*	*PO1*	**10**
	c)	Differentiate the mathematical analysis of non-recursive algorithms from recursive algorithms with suitable examples.	*CO2*	*PO2*	**5**
		OR			
2	a)	Explain the fundamental steps involved in algorithmic problem solving with a suitable example. Also discuss the key characteristics that define a good algorithm.	*CO1*	*PO1*	**10**
	b)	Find algorithm efficiency w.r.t time and space for the following algorithms:	*CO2*	*PO2*	**10**

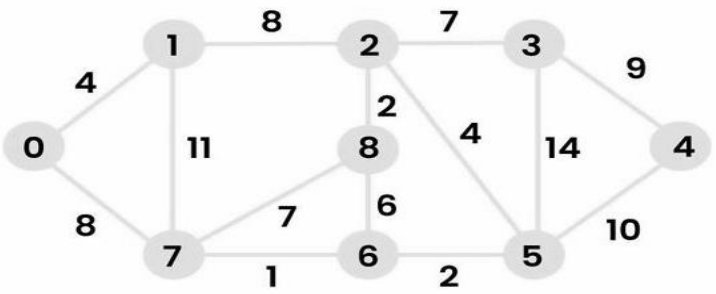
i)
```
Algorithm: CountTriplets(n)
Input: A positive integer n
Output: Total number of
triplets (i, j, k)

Step 1: Initialize count ← 0
Step 2: For i ← 0 to n-1 do
    For j ← 0 to n-1 do
        For k ← 0 to n-1 do
            Increment count by 1
Step 3: Return count
```

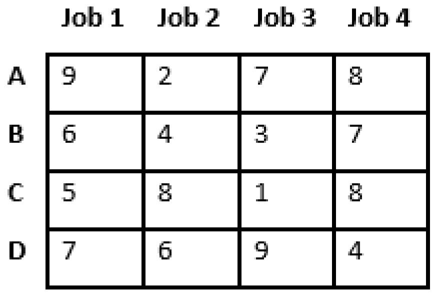
ii)
```
Algo_Hanoi (disk, source, dest, aux)
{
   IF disk == 1, THEN
      move disk from source to dest
   ELSE
      Hanoi (disk - 1, source, aux, dest
     move disk from source to dest
      Hanoi (disk - 1, aux, dest, source)
   END IF
 }
```

			UNIT - II			
	3	a)	Write an algorithm for breadth-first search graph traversal and illustrate the traversal order for the following social network graph, starting with Jim:  Job Jim Kim Tom Bob Cris Ken Ben Ele	*CO1*	*PO1*	**7**
		b)	Write an algorithm for brute-force string matching. Explain with an example string and search pattern.	*CO1*	*PO1*	**7**
		c)	Solve the Travelling Salesman problem using exhaustive search for the given graph. 	*CO1*	*PO1*	**6**
			OR			
	4	a)	Write an algorithm for depth-first search. Elucidate the use of a stack in the traversal of DFS with a graph.	*CO1*	*PO1*	**8**
		b)	The following is the graph of courses offered, with course codes representing the nodes. Identify the appropriate algorithm to illustrate and find an order in which all these courses can be taken.  317 411 510 216 301 518 601 605 142 313 323 622	*CO3*	*PO1*	**8**
		c)	Elucidate any one algorithm in exhaustive search with an example.	*CO1*	*PO1*	**4**

<table>
<tr><td></td><td></td><td></td><td><b>UNIT - III</b></td><td></td><td></td><td></td></tr>
<tr><td></td><td>5</td><td>a)</td><td>Sketch the tracing of sorting the following elements using Heap Sort. Write the algorithm for heapify and heap sort.<table><tr><td><b>23</b></td><td><b>84</b></td><td><b>25</b></td><td><b>34</b></td><td><b>42</b></td><td><b>72</b></td><td><b>11</b></td></tr></table></td><td><i>CO2</i></td><td><i>PO1</i></td><td><b>10</b></td></tr>
<tr><td></td><td></td><td>b)</td><td>Elucidate how Strassen’s matrix multiplication algorithm is more efficient than the traditional matrix multiplication with an example of any two 2 X 2 matrices.</td><td><i>CO1</i></td><td><i>PO2</i></td><td><b>5</b></td></tr>
<tr><td></td><td></td><td>c)</td><td>Differentiate between open hashing and separate chaining.</td><td><i>CO2</i></td><td><i>PO2</i></td><td><b>5</b></td></tr>
<tr><td></td><td></td><td></td><td><b>OR</b></td><td></td><td></td><td></td></tr>
<tr><td></td><td>6</td><td>a)</td><td>Sort the following elements using the Merge sort algorithm that follows the divide and conquer technique. Write a neat sketch of recursive call tracing and write the algorithm.<table><tr><td><b>12</b></td><td><b>91</b></td><td><b>43</b></td><td><b>57</b></td><td><b>8</b></td><td><b>26</b></td><td><b>35</b></td><td><b>88</b></td><td><b>99</b></td><td><b>64</b></td></tr></table></td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>8</b></td></tr>
<tr><td></td><td></td><td>b)</td><td>Write an algorithm for quick sort and compare the time complexity for average, best, and worst case with an example array of elements for each case.</td><td><i>CO1</i></td><td><i>PO2</i></td><td><b>6</b></td></tr>
<tr><td></td><td></td><td>c)</td><td>Illustrate Horspool’s Algorithm for searching a pattern <b>"question"</b> in the string <b>"To be, or not to be, that is the question."</b></td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>6</b></td></tr>
<tr><td></td><td></td><td></td><td><b>UNIT - IV</b></td><td></td><td></td><td></td></tr>
<tr><td></td><td>7</td><td>a)</td><td>Using Floyd’s algorithm, find all pair shortest path for the given graph below:

 </td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>8</b></td></tr>
<tr><td></td><td></td><td>b)</td><td>“A city is planning to optimize its <b>real-time emergency response system</b>. The city map is modeled as a <b>directed weighted graph</b>, where intersections are nodes and roads are edges with weights representing <b>travel times</b> in minutes.”
The control center needs to operate: <b>determine the shortest route from the emergency center (at a fixed node) to any destination node when an incident is reported.</b> For the above operation, you can solve it using Floyd’s or Dijkstra’s algorithms. Which one would you choose to operate? Justify your answer by writing the algorithm.</td><td><i>CO3</i></td><td><i>PO2</i></td><td><b>6</b></td></tr>
<tr><td></td><td></td><td>c)</td><td>Solve the Knapsack problem using Dynamic programming for the items weight = {1,2,1,3} and profits = {10,7,11,15} respectively. The capacity of the sack is 5.</td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>6</b></td></tr>
<tr><td></td><td></td><td></td><td><b>OR</b></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td></td><td>8</td><td>a)</td><td>Find the shortest path from Vertex “0” for the given graph using Dijkstra’s algorithm and write the algorithm.


0
1
2
3
4
5
6
7
8
4
8
7
9
2
11
4
14
7
6
8
10
1
2

</td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>12</b></td></tr>
<tr><td></td><td></td><td>b)</td><td>Find the binomial coefficient C(n, k) where n= 7 and k=5 using dynamic programming.</td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>8</b></td></tr>
<tr><td></td><td></td><td></td><td><b>UNIT - V</b></td><td></td><td></td><td></td></tr>
<tr><td></td><td>9</td><td>a)</td><td>Define NP-Complete problems. Write the significance of NP-Completeness in algorithm analysis. Also, name any two NP-Complete problems.</td><td><i>CO1</i></td><td><i>PO2</i></td><td><b>4</b></td></tr>
<tr><td></td><td></td><td>b)</td><td>Find the optimal solution using the branch and bound technique for the given job assignment problem.
<table><tr><th></th><th>Job 1</th><th>Job 2</th><th>Job 3</th><th>Job 4</th></tr><tr><td>A</td><td>9</td><td>2</td><td>7</td><td>8</td></tr><tr><td>B</td><td>6</td><td>4</td><td>3</td><td>7</td></tr><tr><td>C</td><td>5</td><td>8</td><td>1</td><td>8</td></tr><tr><td>D</td><td>7</td><td>6</td><td>9</td><td>4</td></tr></table></td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>8</b></td></tr>
<tr><td></td><td></td><td>c)</td><td>Write the state-space tree can be constructed for the following instances S = (3, 5, 6, 7) and d = 15 using Sum of Subsets algorithm.</td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>8</b></td></tr>
<tr><td></td><td></td><td></td><td><b>OR</b></td><td></td><td></td><td></td></tr>
<tr><td></td><td>10</td><td>a)</td><td>Differentiate between P and NP problems.</td><td><i>CO1</i></td><td><i>PO2</i></td><td><b>6</b></td></tr>
<tr><td></td><td></td><td>b)</td><td>Find any <b>one</b> solution for the 4-queens problem and write the state space tree.</td><td><i>CO1</i></td><td><i>PO1</i></td><td><b>6</b></td></tr>
<tr><td></td><td></td><td>c)</td><td>Elicit the advantage of solving the travelling salesman problem using branch and bound over the brute force technique with an example.</td><td><i>CO2</i></td><td><i>PO2</i></td><td><b>8</b></td></tr>
</table>

******